WHAT IS CLAIMED IS:

1. A method comprising:

receiving a call to iterate through a collection including at least one uninstantiated element;

after receiving the call, instantiating the uninstantiated element to provide an instantiated element; and

iterating through the instantiated element.

2. The method of claim 1, further comprising:

implementing an interface having routines for iterating through the collection.

3. The method of claim 1, further comprising:

implementing an interface having routines for instantiating the uninstantiated element based on the format of the uninstantiated element.

4. The method of claim 1, further comprising:

determining whether the uninstantiated element is available in the collection.

5. The method of claim 1, wherein the collection is a file and the uninstantiated element is a file message.

6. The method of claim 1, wherein the collection is a web page and the uninstantiated element is a web link.

7. The method of claim 1, wherein the collection is an SQL database table and the uninstantiated element is a database field.

8. A method comprising:

after receiving a call to iterate through raw data, generating data elements from the raw data, wherein the data elements can be iterated through; and

iterating through the data elements.

9. The method of claim 8, wherein the data elements are generated based on the format of the raw data.

10. A method comprising:

receiving an instruction to iterate through a file including at least one message;

determining to which of a plurality of predefined formats the message belongs;

instantiating the message using a routine associated with the determined format of the message; and

iterating through the instantiated message.

11. The method of claim 10, further comprising:

determining whether the file includes subsequent messages to be retrieved; and if so,

retrieving a next message using a routine associated with the determined format, and

optionally, removing the next message from the file.

12. The method of claim 10, further comprising:

opening the file;

determining whether the message is available from the file; and

closing the file after the message has been retrieved.

13. The method of claim 10, wherein the predefined formats are transparent to a user.

14. The method of claim 10, wherein the predefined formats include a fixed length header and data bytes.

15. The method of claim 14, wherein the routine includes:

reading the fixed length header from the file;

calculating an integer value of the fixed length header;

reading the data bytes disposed in the file after the fixed length header, the number of read data bytes corresponding to the integer value; and

returning the read data bytes as the message.

16. The method of claim 10, wherein the predefined formats include delimiters separating data bytes.

17. The method of claim 16, wherein the routine includes:

reading the data bytes until a delimiter is reached; and

returning the read data bytes as the message.

18. The method of claim 10, further comprising:

receiving an instruction to iterate through an empty file; and

returning an indication that the empty file does not include any messages.

19. A machine readable medium containing program instructions for execution on a processor, which when executed by the processor, cause the processor to perform:

calling a message reader object to iterate through a collection;

creating a file message reader object to determine the format of an element in the collection;

using the file message reader object to read the element from the collection; and

after the calling the message reader object, using the file message reader object to create a message object from the read element.

20. The machine readable medium of claim 19, wherein the file message reader object includes a routine to retrieve the element from the collection and create the message object from the retrieved element.

21. The machine readable medium of claim 19, wherein the message iterator object includes a first routine to create the file message reader object and a second routine to iterate through the collection.

22. The machine readable medium of claim 19, wherein the format includes a fixed length header and data bytes.

23. The machine readable medium of claim 19, wherein the format includes delimiters separating data bytes.

24. The machine readable medium of claim 19, further comprising:

determining the format of the element during the execution by reading at least a portion of the element and evaluating the portion.

25. A method comprising:

responsive to a request to read an element from a collection, creating a reader object based on the format of the element;

opening the collection;

after receiving the request,

using the reader object, instantiating an element object including the element, and

performing an operation on the element object; and

closing the collection.

26. The method of claim 25, further comprising:

repeating the instantiating the element object and performing if there are additional elements available to be read from the collection.

27. The method of claim 25, further comprising:

removing the element from the collection after the instantiating or the performing.